FAX

RECEIVED
CENTRAL FAX CENTER
SEP 2 0 2005

ATTN. Cam Linh T. Nguyen

Fax Number 1 571 273 8300

Phone Number 571 272 4024

FROM Volel Emile, Esq.

Fax Number (512) 306-0240

Phone Number (512) 306-7969

SUBJECT Response to Non-Compliant Appeal Brief

Number of Pages 39

Date 9/20/2005

MESSAGE

This fax transmission includes:

PTO/SB/21 (02-04)
Approved for use through 07/31/2006. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Application Number	10/099,777
Filing Date	03/14/2002
First Named Inventor	Brown et al.
Art Unit	2171
Examiner Name	Cam Linh T. Nguyen
Attorney Docket Number	AUS920010886US1

Total Number of Pages in This Submission

ENCLOSURES (Check all that apply)

- [x] Fee Transmittal Form
- [x] Fee Attached
- [ ] Amendment/Reply
- [ ] After Final
- [ ] Affidavits/declaration(s)
- [x] Extension of Time Request
- [ ] Express Abandonment Request
- [ ] Information Disclosure Statement
- [ ] Certified Copy of Priority Document(s)
- [ ] Response to Missing Parts/ Incomplete Application
- [ ] Response to Missing Parts under 37 CFR 1.52 or 1.53
- [ ] Drawing(s)
- [ ] Licensing-related Papers
- [ ] Petition
- [ ] Petition to Convert to a Provisional Application
- [ ] Power of Attorney, Revocation Change of Correspondence Address
- [ ] Terminal Disclaimer
- [ ] Request for Refund
- [ ] CD, Number of CD(s) _____
- [ ] After Allowance communication to Technology Center (TC)
- [ ] Appeal Communication to Board of Appeals and Interferences
- [x] Appeal Communication to TC (Appeal Notice, Brief, Reply Brief)
- [ ] Proprietary Information
- [ ] Status Letter
- [ ] Other Enclosure(s) (please identify below):

Remarks

One month extension of time to file a Response to a Non-Compliant Appeal Brief Action.

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm or Individual name	Volel Emile
Signature	
Date	09/20/2005

CERTIFICATE OF TRANSMISSION/MAILING

I hereby certify that this correspondence is being facsimile transmitted to the USPTO or deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on the date shown below.

Typed or printed name	Volel Emile		
Signature		Date	09/20/2005

RECEIVED
CENTRAL FAX CENTER
SEP 20 2005

PTO/SB/17 (12-04v2)
Approved for use through 07/31/2006. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

Effective on 12/08/2004.
Fees pursuant to the Consolidated Appropriations Act, 2005 (H.R. 4818).

FEE TRANSMITTAL For FY 2005

☐ Applicant claims small entity status. See 37 CFR 1.27

TOTAL AMOUNT OF PAYMENT ($) 120.00

Complete if Known	
Application Number	10/099,777
Filing Date	03/14/2002
First Named Inventor	William B. Brown
Examiner Name	Cam Linh T. Nguyen
Art Unit	2171
Attorney Docket No.	AUS920010866US1

METHOD OF PAYMENT (check all that apply)

☐ Check ☑ Credit Card ☐ Money Order ☐ None ☐ Other (please identify): _____

☐ Deposit Account Deposit Account Number: _____ Deposit Account Name: _____

For the above-identified deposit account, the Director is hereby authorized to: (check all that apply)

☐ Charge fee(s) indicated below ☐ Charge fee(s) indicated below, except for the filing fee

☐ Charge any additional fee(s) or underpayments of fee(s) under 37 CFR 1.16 and 1.17 ☐ Credit any overpayments

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

FEE CALCULATION

1. BASIC FILING, SEARCH, AND EXAMINATION FEES

Application Type	FILING FEES Fee ($)	FILING FEES Small Entity Fee ($)	SEARCH FEES Fee ($)	SEARCH FEES Small Entity Fee ($)	EXAMINATION FEES Fee ($)	EXAMINATION FEES Small Entity Fee ($)	Fees Paid ($)
Utility	300	150	500	250	200	100	
Design	200	100	100	50	130	65	
Plant	200	100	300	150	160	80	
Reissue	300	150	500	250	600	300	
Provisional	200	100	0	0	0	0	

2. EXCESS CLAIM FEES

Fee Description	Fee ($)	Small Entity Fee ($)
Each claim over 20 (including Reissues)	50	25
Each independent claim over 3 (including Reissues)	200	100
Multiple dependent claims	360	180

Total Claims	Extra Claims	Fee ($)	Fee Paid ($)
_____ - 20 or HP =	_____	x _____	= _____

HP = highest number of total claims paid for, if greater than 20.

Indep. Claims	Extra Claims	Fee ($)	Fee Paid ($)
_____ - 3 or HP =	_____	x _____	= _____

HP = highest number of independent claims paid for, if greater than 3.

Multiple Dependent Claims Fee ($)	Fee Paid ($)
_____	_____

3. APPLICATION SIZE FEE

If the specification and drawings exceed 100 sheets of paper (excluding electronically filed sequence or computer listings under 37 CFR 1.52(e)), the application size fee due is $250 ($125 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).

Total Sheets	Extra Sheets	Number of each additional 50 or fraction thereof	Fee ($)	Fee Paid ($)
_____ - 100 =	_____	/ 50 = _____ (round up to a whole number)	x _____	= _____

4. OTHER FEE(S)

	Fees Paid ($)
Non-English Specification, $130 fee (no small entity discount)	
Other (e.g., late filing surcharge): One Month Extension of Time Fee.	120.00

SUBMITTED BY

Signature		Registration No. (Attorney/Agent) 39,969	Telephone 512 306 7969
Name (Print/Type)	Volel Emile		Date 09/20/2005

RECEIVED
CENTRAL FAX CENTER

SEP 2 0 2005

Appl. No. 10/099,777
Substitute Appeal Brief dated 09/20/2005
Reply to Office Action of 07/28/2005

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re: Application of: Brown et al.	
Serial No: 10/099,777	Before the Examiner: Cam Linh T Nguyen
Filed: 03/14/2002	Group Art Unit: 2171
Title: APPARATUS AND METHOD OF EXPORTING FILE SYSTEMS WITHOUT FIRST MOUNTING THE FILE SYSTEMS	Confirmation No.: 4836

TRANSMITTAL OF APPELLANTS' RESPONSE OF NON-COMPLIANT APPEAL BRIEF UNDER 37 C.F.R. 41.37

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Attached is Appellants' Response to a Notification of Non-compliant Appeal Brief dated **07/28/2005**. The Response is in triplicate.

The item(s) marked below is (are) appropriate:

1. __X_ A fee for a one-month extension of term for reply to the Notification of Non-compliance Appeal Brief is attached thereto.
2. ____ Appeal fee

   ____ other than a small entity.
3. ____ Payment

   ____ Please charge Deposit Account **09-0447** the sum of $500.00. A duplicate of this notice is attached.

Appl. No. 10/099,777
Substitute Appeal Brief dated 09/20/2005
Reply to Office Action of 07/28/2005

The Commissioner is hereby authorized to charge any additional fee, which may be required or credit any overpayment to Deposit Account No. **09-0447**.

Respectfully submitted,

Volel Emile
Attorney for Applicants
Registration No. 39,969
(512) 306-7969

RECEIVED
CENTRAL FAX CENTER

SEP 2 0 2005

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re: Application of: Brown et al.	
Serial No: 10/099,777	Before the Examiner: Cam Linh T Nguyen
Filed: 03/14/2002	Group Art Unit: 2171
Title: APPARATUS AND METHOD OF EXPORTING FILE SYSTEMS WITHOUT FIRST MOUNTING THE FILE SYSTEMS	Confirmation No.: 4836

RESPONSE TO NOTIFICATION OF NON-COMPLIANT APPEAL BRIEF UNDER 37 C.F.R. 41.37

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This substitute Appeal Brief is being submitted in response to the NOTIFICATION OF NON-COMPLIANT APPEAL BRIEF of July 28, 2005.

09/21/2005 TL0111 00000023 10099777
01 FC:1251 120.00 OP

BRIEF FOR APPLICANTS - APPELLANTS

(1)

Real Party in Interest

The real party in interest is International Business Machines Corporation (IBM), the assignee.

(2)

Related Appeals and Interferences

There are no other appeals or interferences known to appellants, appellants' representative or assignee, which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3)

Status of Claims

Claims 1 - 20 have been finally rejected. This appeal involves all the rejected claims.

(4)

Status of Amendment

A Response to the first Office Action, in which the independent claims (Claims 1, 6, 11 and 16) were amended, was filed on December 13, 2004. The Amendment was entered; however, the Examiner did not find Applicants' arguments to be persuasive and issued a Final Office Action on March 8, 2005.

(5)

Summary of the Invention

The present invention provides a method of exporting file systems. According to the invention, a file in which all information needed to mount file systems at a particular mount point is associated with the mount point (see page 14, lines 22 - 24 and page 15, lines 18 - 20). Thus, when a file is to be exported, the file is consulted to retrieve the information needed to mount the file system. Once the information is retrieved, the file is exported (see page 15, lines 21 - 25 and page 16, lines 9 - 12 as well as originally filed Fig. 8).

(6)

Issues

Whether claims 1 - 20 were properly rejected under 102(a) as being anticipated by Vahalia et al.

(7)

Grouping of Claims

The rejected claims stand or fall together.

(8)

Argument

In considering a Section 102 rejection, all the elements of the claimed invention must be disclosed in a single item of prior art in the form literally defined in the claim. *Jamesbury Corp. v. Litton Indus. Products*, 756 F.2d 1556, 225 USPQ 253 (Fed. Cir. 1985); *Atlas Powder Co. v. Dupont*, 750 F.2d 1569, 224 USPQ 409 (Fed. Cir. 1984); *American Hospital Supply v. Travenol Labs.*, 745 F.2d 1, 223 USPQ 577 (Fed. Cir. 1984). Russell-Falla et al., the

reference used to reject the independent claims, does not disclose all the elements of the claims.

Vahalia et al. purport to teach a method of: (1) providing NFS clients with read/write access to read from and write into file systems; and (2) performing failure recovery of a failed server.

In the method of providing NFS clients with read/write access to read from and write into file systems, each file system is assigned to a particular server in a network of servers. Any server in the network may receive a file access request from any NFS client. If the file system that is to be accessed is assigned to the server that receives the request, that server will provide the access. But, if the file system that is to be accessed is not assigned to the server that receives the request, the server will forward the request to the server to which the file system is assigned.

This scheme obviates the need to provide coherency since only the server to which the file is assigned will allow changes to any file in the file system and will presumably permit only one client to make changes to the files in the file system at a time. Further, the scheme provides a certain level of load balancing as only a server to which a file system is assigned will process the request and presumably access requests will be sent to different file systems assigned to different servers in the network.

To determine to which one of the servers the file system is assigned, a file that contains file system/computer assignment information is consulted.

In the method of performing failure recovery of a failed server, one of the servers monitors the rest of the servers to detect failures. When a failure of a server is detected, the file systems that were assigned to that failed server are re-assigned to an operational server. Thus, requests can always be processed.

Since to export a file system is to make the file system available for NFS clients to mount (an NFS client can only mount a file system after the file system has been exported to it) and since the NFS clients disclosed by Vahalia et al. are requesting access to a file system (an NFS client cannot request access to a file system unless and until the file system is mounted on the client), ***Vahalia et al. do not teach, show or suggest a method of exporting file systems*** as stated by the examiner.

Put differently, in order for an NFS client to request access to a file system, the file system must have already been mounted on the NFS client, which means that the file system must have already been exported to the NFS client. The disclosure of Vahalia et al. does not discuss file system exportation but delves straight into file systems access request grants.

The Examiner cited different passages in col. 13 of the disclosure of Vahalia et al. to support the rejection. However, the cited passages merely explain the disclosed method.

Firstly, it is well known that NFS files are indexed in a file directory that may be said to be organized as a tree, and each file system may be identified by a node in

the tree. It is further well known that files are mounted to the "tree" at specific points called "mount points".

Vahalia et al. disclose in col. 13, lines 19 - 22 that each computer has (1) a directory of the file systems; (2) a database of the mount points for the file systems and (3) the computer to which each read-write file system is assigned.

Vahalia et al. further disclose in col. 13, line 40 to col. 14, line 15 that item 3 above allows a computer which receives a request to access a file system from an NFS client to check to see whether the file system is assigned to it or to another computer (see specifically col. 13, lines 22 - 26). If the file system is assigned to another computer, then the computer that receives the request may forward the request to the other computer. If the file system is assigned to the computer that receives the request, then it needs to know whether the file system is remote or not. As it is defined in the reference, a remote file system is a file system that has to be mounted on another file system (see specifically col. 13, lines 52 - 55 and col. 17, lines 15 - 32). To determine whether the file system is remote, the name (i.e., pathname) of the file that is to be accessed in the file system is parsed. If a mount point is reached as indicated by the list of mount points in the database in (2), then it is remote. If the file system to be accessed is remote, then a request to mount it at the proper mount point will be issued to the computer that has the file system to which it (the file system that is to be accessed) is to be mounted. If the file system is not remote, then before granting the request

the computer that receives the request will make sure that the file system to be accessed was exported to the NFS client that is issued the file access request (after all, if the file system was not exported to the client requesting access then the request must be in error since that client should not even know of the file system).

Thus, in column 13, which was extensively cited by the Examiner as support for the rejection, Vahalia et al. merely explain a specific implementation of their method. However, nowhere in that implementation is there disclosed the claimed invention.

In other words, Vahalia et al. do not teach, show or suggest ***consulting a file associated with a mount point of a mounted file system to retrieve information needed to export file systems*** that are to be mounted at that mount point as claimed.

Therefore, Applicants submit that the claims in the Application should be allowable. Hence, Applicants respectfully request allowance and passage to issue of the claims in the application.

Respectfully submitted,

By: _____
Volel Emile
Attorney for Applicants
Registration No. 39,969
(512) 306-7969

Appendix

1. (Previously presented) A method of exporting file systems comprising the steps of:

consulting a file associated with a mount point of a mounted file system to retrieve needed information to export the file systems, the mount point being the point at which the file systems are mounted on a computer system; and

exporting the file systems.

2. (Original) The method of Claim 1 wherein the needed information is names of devices within which the file systems are located.

3. (Original) The method of Claim 2 wherein the file systems are exported without first being mounted.

4. (Original) The method of Claim 3 wherein the file is an extended attribute file.

5. (Original) The method of Claim 4 wherein each mount point has an extended attribute file.

6. (Previously presented) A computer program product on a computer readable medium for exporting file systems comprising:

code means for consulting a file associated with a mount point of a mounted file system to retrieve needed information to export the file systems, the mount point being the point at which the file systems are mounted on a computer system; and

code means for exporting the file systems.

7. (Original) The computer program product of Claim 6 wherein the needed information is names of devices within which the file systems are located.

8. (Original) The computer program product of Claim 7 wherein the file systems are exported without first being mounted.

9. (Original) The computer program product of Claim 8 wherein the file is an extended attribute file.

10. (Original) The computer program product of Claim 9 wherein each mount point has an extended attribute file.

11. (Previously presented) An apparatus for exporting file systems comprising:

means for consulting a file associated with a mount point of a mounted file system to retrieve needed information to export the file systems, the mount

point being the point at which the file systems are mounted on a computer system; and

means for exporting the file systems.

12. (Original) The apparatus of Claim 11 wherein the needed information is names of devices within which the file systems are located.

13. (Original) The apparatus of Claim 12 wherein the file systems are exported without first being mounted.

14. (Original) The apparatus of Claim 13 wherein the file is an extended attribute file.

15. (Original) The apparatus of Claim 14 wherein each mount point has an extended attribute file.

16. (Previously presented) A computer system for exporting file systems comprising:

at least one storage device for storing code data; and

at least one processor for processing the code data to consult a file associated with a mount point of a mounted file system to retrieve needed information to export the file systems, the mount point being the point at which the file systems are mounted on ~~a~~ the computer system, and to export the file systems.

17. (Original) The computer system of Claim 16 wherein the needed information is names of devices within which the file systems are located.

18. (Original) The computer system of Claim 17 wherein the file systems are exported without first being mounted.

19. (Original) The computer system of Claim 16 wherein the file is an extended attribute file.

20. (Original) The computer system of Claim 19 wherein each mount point has an extended attribute file.

RECEIVED
CENTRAL FAX CENTER

SEP 2 0 2005

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re: Application of: Brown et al.	
Serial No: 10/099,777	Before the Examiner: Cam Linh T Nguyen
Filed: 03/14/2002	Group Art Unit: 2171
Title: APPARATUS AND METHOD OF EXPORTING FILE SYSTEMS WITHOUT FIRST MOUNTING THE FILE SYSTEMS	Confirmation No.: 4836

RESPONSE TO NOTIFICATION OF NON-COMPLIANT APPEAL BRIEF UNDER 37 C.F.R. 41.37

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This substitute Appeal Brief is being submitted in response to the NOTIFICATION OF NON-COMPLIANT APPEAL BRIEF of July 28, 2005.

BRIEF FOR APPLICANTS - APPELLANTS

(1)

Real Party in Interest

The real party in interest is International Business Machines Corporation (IBM), the assignee.

(2)

Related Appeals and Interferences

There are no other appeals or interferences known to appellants, appellants' representative or assignee, which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3)

Status of Claims

Claims 1 - 20 have been finally rejected. This appeal involves all the rejected claims.

(4)

Status of Amendment

A Response to the first Office Action, in which the independent claims (Claims 1, 6, 11 and 16) were amended, was filed on December 13, 2004. The Amendment was entered; however, the Examiner did not find Applicants' arguments to be persuasive and issued a Final Office Action on March 8, 2005.

(5)

Summary of the Invention

The present invention provides a method of exporting file systems. According to the invention, a file in which all information needed to mount file systems at a particular mount point is associated with the mount point (see page 14, lines 22 - 24 and page 15, lines 18 - 20). Thus, when a file is to be exported, the file is consulted to retrieve the information needed to mount the file system. Once the information is retrieved, the file is exported (see page 15, lines 21 - 25 and page 16, lines 9 - 12 as well as originally filed Fig. 8).

(6)

Issues

Whether claims 1 - 20 were properly rejected under 102(a) as being anticipated by Vahalia et al.

(7)

Grouping of Claims

The rejected claims stand or fall together.

(8)

Argument

In considering a Section 102 rejection, all the elements of the claimed invention must be disclosed in a single item of prior art in the form literally defined in the claim. *Jamesbury Corp. v. Litton Indus. Products*, 756 F.2d 1556, 225 USPQ 253 (Fed. Cir. 1985); *Atlas Powder Co. v. Dupont*, 750 F.2d 1569, 224 USPQ 409 (Fed. Cir. 1984); *American Hospital Supply v. Travenol Labs.*, 745 F.2d 1, 223 USPQ 577 (Fed. Cir. 1984). Russell-Falla et al., the

reference used to reject the independent claims, does not disclose all the elements of the claims.

Vahalia et al. purport to teach a method of: (1) providing NFS clients with read/write access to read from and write into file systems; and (2) performing failure recovery of a failed server.

In the method of providing NFS clients with read/write access to read from and write into file systems, each file system is assigned to a particular server in a network of servers. Any server in the network may receive a file access request from any NFS client. If the file system that is to be accessed is assigned to the server that receives the request, that server will provide the access. But, if the file system that is to be accessed is not assigned to the server that receives the request, the server will forward the request to the server to which the file system is assigned.

This scheme obviates the need to provide coherency since only the server to which the file is assigned will allow changes to any file in the file system and will presumably permit only one client to make changes to the files in the file system at a time. Further, the scheme provides a certain level of load balancing as only a server to which a file system is assigned will process the request and presumably access requests will be sent to different file systems assigned to different servers in the network.

To determine to which one of the servers the file system is assigned, a file that contains file system/computer assignment information is consulted.

In the method of performing failure recovery of a failed server, one of the servers monitors the rest of the servers to detect failures. When a failure of a server is detected, the file systems that were assigned to that failed server are re-assigned to an operational server. Thus, requests can always be processed.

Since to export a file system is to make the file system available for NFS clients to mount (an NFS client can only mount a file system after the file system has been exported to it) and since the NFS clients disclosed by Vahalia et al. are requesting access to a file system (an NFS client cannot request access to a file system unless and until the file system is mounted on the client), ***Vahalia et al. do not teach, show or suggest a method of exporting file systems*** as stated by the examiner.

Put differently, in order for an NFS client to request access to a file system, the file system must have already been mounted on the NFS client, which means that the file system must have already been exported to the NFS client. The disclosure of Vahalia et al. does not discuss file system exportation but delves straight into file systems access request grants.

The Examiner cited different passages in col. 13 of the disclosure of Vahalia et al. to support the rejection. However, the cited passages merely explain the disclosed method.

Firstly, it is well known that NFS files are indexed in a file directory that may be said to be organized as a tree, and each file system may be identified by a node in

the tree. It is further well known that files are mounted to the "tree" at specific points called "mount points".

Vahalia et al. disclose in col. 13, lines 19 - 22 that each computer has (1) a directory of the file systems; (2) a database of the mount points for the file systems and (3) the computer to which each read-write file system is assigned.

Vahalia et al. further disclose in col. 13, line 40 to col. 14, line 15 that item 3 above allows a computer which receives a request to access a file system from an NFS client to check to see whether the file system is assigned to it or to another computer (see specifically col. 13, lines 22 - 26). If the file system is assigned to another computer, then the computer that receives the request may forward the request to the other computer. If the file system is assigned to the computer that receives the request, then it needs to know whether the file system is remote or not. As it is defined in the reference, a remote file system is a file system that has to be mounted on another file system (see specifically col. 13, lines 52 - 55 and col. 17, lines 15 - 32). To determine whether the file system is remote, the name (i.e., pathname) of the file that is to be accessed in the file system is parsed. If a mount point is reached as indicated by the list of mount points in the database in (2), then it is remote. If the file system to be accessed is remote, then a request to mount it at the proper mount point will be issued to the computer that has the file system to which it (the file system that is to be accessed) is to be mounted. If the file system is not remote, then before granting the request

the computer that receives the request will make sure that the file system to be accessed was exported to the NFS client that is issued the file access request (after all, if the file system was not exported to the client requesting access then the request must be in error since that client should not even know of the file system).

Thus, in column 13, which was extensively cited by the Examiner as support for the rejection, Vahalia et al. merely explain a specific implementation of their method. However, nowhere in that implementation is there disclosed the claimed invention.

In other words, Vahalia et al. do not teach, show or suggest ***consulting a file associated with a mount point of a mounted file system to retrieve information needed to export file systems*** that are to be mounted at that mount point as claimed.

Therefore, Applicants submit that the claims in the Application should be allowable. Hence, Applicants respectfully request allowance and passage to issue of the claims in the application.

Respectfully submitted,

By: _____
Volel Emile
Attorney for Applicants
Registration No. 39,969
(512) 306-7969

Appendix

1. (Previously presented) A method of exporting file systems comprising the steps of:

consulting a file associated with a mount point of a mounted file system to retrieve needed information to export the file systems, the mount point being the point at which the file systems are mounted on a computer system; and

exporting the file systems.

2. (Original) The method of Claim 1 wherein the needed information is names of devices within which the file systems are located.

3. (Original) The method of Claim 2 wherein the file systems are exported without first being mounted.

4. (Original) The method of Claim 3 wherein the file is an extended attribute file.

5. (Original) The method of Claim 4 wherein each mount point has an extended attribute file.

6. (Previously presented) A computer program product on a computer readable medium for exporting file systems comprising:

code means for consulting a file associated with a mount point of a mounted file system to retrieve needed information to export the file systems, the mount point being the point at which the file systems are mounted on a computer system; and

code means for exporting the file systems.

7. (Original) The computer program product of Claim 6 wherein the needed information is names of devices within which the file systems are located.

8. (Original) The computer program product of Claim 7 wherein the file systems are exported without first being mounted.

9. (Original) The computer program product of Claim 8 wherein the file is an extended attribute file.

10. (Original) The computer program product of Claim 9 wherein each mount point has an extended attribute file.

11. (Previously presented) An apparatus for exporting file systems comprising:

means for consulting a file associated with a mount point of a mounted file system to retrieve needed information to export the file systems, the mount

point being the point at which the file systems are mounted on a computer system; and

means for exporting the file systems.

12. (Original) The apparatus of Claim 11 wherein the needed information is names of devices within which the file systems are located.

13. (Original) The apparatus of Claim 12 wherein the file systems are exported without first being mounted.

14. (Original) The apparatus of Claim 13 wherein the file is an extended attribute file.

15. (Original) The apparatus of Claim 14 wherein each mount point has an extended attribute file.

16. (Previously presented) A computer system for exporting file systems comprising:

at least one storage device for storing code data; and

at least one processor for processing the code data to consult a file associated with a mount point of a mounted file system to retrieve needed information to export the file systems, the mount point being the point at which the file systems are mounted on ~~a~~ the computer system, and to export the file systems.

17. (Original) The computer system of Claim 16 wherein the needed information is names of devices within which the file systems are located.

18. (Original) The computer system of Claim 17 wherein the file systems are exported without first being mounted.

19. (Original) The computer system of Claim 16 wherein the file is an extended attribute file.

20. (Original) The computer system of Claim 19 wherein each mount point has an extended attribute file.

RECEIVED
CENTRAL FAX CENTER

SEP 2 0 2005

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re: Application of: Brown et al.	
Serial No: 10/099,777	Before the Examiner: Cam Linh T Nguyen
Filed: 03/14/2002	Group Art Unit: 2171
Title: APPARATUS AND METHOD OF EXPORTING FILE SYSTEMS WITHOUT FIRST MOUNTING THE FILE SYSTEMS	Confirmation No.: 4836

RESPONSE TO NOTIFICATION OF NON-COMPLIANT APPEAL BRIEF UNDER 37 C.F.R. 41.37

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This substitute Appeal Brief is being submitted in response to the NOTIFICATION OF NON-COMPLIANT APPEAL BRIEF of July 28, 2005.

BRIEF FOR APPLICANTS - APPELLANTS

(1)

Real Party in Interest

The real party in interest is International Business Machines Corporation (IBM), the assignee.

(2)

Related Appeals and Interferences

There are no other appeals or interferences known to appellants, appellants' representative or assignee, which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3)

Status of Claims

Claims 1 - 20 have been finally rejected. This appeal involves all the rejected claims.

(4)

Status of Amendment

A Response to the first Office Action, in which the independent claims (Claims 1, 6, 11 and 16) were amended, was filed on December 13, 2004. The Amendment was entered; however, the Examiner did not find Applicants' arguments to be persuasive and issued a Final Office Action on March 8, 2005.

(5)

Summary of the Invention

The present invention provides a method of exporting file systems. According to the invention, a file in which all information needed to mount file systems at a particular mount point is associated with the mount point (see page 14, lines 22 - 24 and page 15, lines 18 - 20). Thus, when a file is to be exported, the file is consulted to retrieve the information needed to mount the file system. Once the information is retrieved, the file is exported (see page 15, lines 21 - 25 and page 16, lines 9 - 12 as well as originally filed Fig. 8).

(6)

Issues

Whether claims 1 - 20 were properly rejected under 102(a) as being anticipated by Vahalia et al.

(7)

Grouping of Claims

The rejected claims stand or fall together.

(8)

Argument

In considering a Section 102 rejection, all the elements of the claimed invention must be disclosed in a single item of prior art in the form literally defined in the claim. *Jamesbury Corp. v. Litton Indus. Products*, 756 F.2d 1556, 225 USPQ 253 (Fed. Cir. 1985); *Atlas Powder Co. v. Dupont*, 750 F.2d 1569, 224 USPQ 409 (Fed. Cir. 1984); *American Hospital Supply v. Travenol Labs.*, 745 F.2d 1, 223 USPQ 577 (Fed. Cir. 1984). Russell-Falla et al., the

reference used to reject the independent claims, does not disclose all the elements of the claims.

Vahalia et al. purport to teach a method of: (1) providing NFS clients with read/write access to read from and write into file systems; and (2) performing failure recovery of a failed server.

In the method of providing NFS clients with read/write access to read from and write into file systems, each file system is assigned to a particular server in a network of servers. Any server in the network may receive a file access request from any NFS client. If the file system that is to be accessed is assigned to the server that receives the request, that server will provide the access. But, if the file system that is to be accessed is not assigned to the server that receives the request, the server will forward the request to the server to which the file system is assigned.

This scheme obviates the need to provide coherency since only the server to which the file is assigned will allow changes to any file in the file system and will presumably permit only one client to make changes to the files in the file system at a time. Further, the scheme provides a certain level of load balancing as only a server to which a file system is assigned will process the request and presumably access requests will be sent to different file systems assigned to different servers in the network.

To determine to which one of the servers the file system is assigned, a file that contains file system/computer assignment information is consulted.

In the method of performing failure recovery of a failed server, one of the servers monitors the rest of the servers to detect failures. When a failure of a server is detected, the file systems that were assigned to that failed server are re-assigned to an operational server. Thus, requests can always be processed.

Since to export a file system is to make the file system available for NFS clients to mount (an NFS client can only mount a file system after the file system has been exported to it) and since the NFS clients disclosed by Vahalia et al. are requesting access to a file system (an NFS client cannot request access to a file system unless and until the file system is mounted on the client), ***Vahalia et al. do not teach, show or suggest a method of exporting file systems*** as stated by the examiner.

Put differently, in order for an NFS client to request access to a file system, the file system must have already been mounted on the NFS client, which means that the file system must have already been exported to the NFS client. The disclosure of Vahalia et al. does not discuss file system exportation but delves straight into file systems access request grants.

The Examiner cited different passages in col. 13 of the disclosure of Vahalia et al. to support the rejection. However, the cited passages merely explain the disclosed method.

Firstly, it is well known that NFS files are indexed in a file directory that may be said to be organized as a tree, and each file system may be identified by a node in

the tree. It is further well known that files are mounted to the "tree" at specific points called "mount points".

Vahalia et al. disclose in col. 13, lines 19 - 22 that each computer has (1) a directory of the file systems; (2) a database of the mount points for the file systems and (3) the computer to which each read-write file system is assigned.

Vahalia et al. further disclose in col. 13, line 40 to col. 14, line 15 that item 3 above allows a computer which receives a request to access a file system from an NFS client to check to see whether the file system is assigned to it or to another computer (see specifically col. 13, lines 22 - 26). If the file system is assigned to another computer, then the computer that receives the request may forward the request to the other computer. If the file system is assigned to the computer that receives the request, then it needs to know whether the file system is remote or not. As it is defined in the reference, a remote file system is a file system that has to be mounted on another file system (see specifically col. 13, lines 52 - 55 and col. 17, lines 15 - 32). To determine whether the file system is remote, the name (i.e., pathname) of the file that is to be accessed in the file system is parsed. If a mount point is reached as indicated by the list of mount points in the database in (2), then it is remote. If the file system to be accessed is remote, then a request to mount it at the proper mount point will be issued to the computer that has the file system to which it (the file system that is to be accessed) is to be mounted. If the file system is not remote, then before granting the request

the computer that receives the request will make sure that the file system to be accessed was exported to the NFS client that is issued the file access request (after all, if the file system was not exported to the client requesting access then the request must be in error since that client should not even know of the file system).

Thus, in column 13, which was extensively cited by the Examiner as support for the rejection, Vahalia et al. merely explain a specific implementation of their method. However, nowhere in that implementation is there disclosed the claimed invention.

In other words, Vahalia et al. do not teach, show or suggest ***consulting a file associated with a mount point of a mounted file system to retrieve information needed to export file systems*** that are to be mounted at that mount point as claimed.

Therefore, Applicants submit that the claims in the Application should be allowable. Hence, Applicants respectfully request allowance and passage to issue of the claims in the application.

Respectfully submitted,

By: _____
Volel Emile
Attorney for Applicants
Registration No. 39,969
(512) 306-7969

Appendix

1. (Previously presented) A method of exporting file systems comprising the steps of:

consulting a file associated with a mount point of a mounted file system to retrieve needed information to export the file systems, the mount point being the point at which the file systems are mounted on a computer system; and

exporting the file systems.

2. (Original) The method of Claim 1 wherein the needed information is names of devices within which the file systems are located.

3. (Original) The method of Claim 2 wherein the file systems are exported without first being mounted.

4. (Original) The method of Claim 3 wherein the file is an extended attribute file.

5. (Original) The method of Claim 4 wherein each mount point has an extended attribute file.

6. (Previously presented) A computer program product on a computer readable medium for exporting file systems comprising:

code means for consulting a file associated with a mount point of a mounted file system to retrieve needed information to export the file systems, the mount point being the point at which the file systems are mounted on a computer system; and

code means for exporting the file systems.

7. (Original) The computer program product of Claim 6 wherein the needed information is names of devices within which the file systems are located.

8. (Original) The computer program product of Claim 7 wherein the file systems are exported without first being mounted.

9. (Original) The computer program product of Claim 8 wherein the file is an extended attribute file.

10. (Original) The computer program product of Claim 9 wherein each mount point has an extended attribute file.

11. (Previously presented) An apparatus for exporting file systems comprising:

means for consulting a file associated with a mount point of a mounted file system to retrieve needed information to export the file systems, the mount

point being the point at which the file systems are mounted on a computer system; and

means for exporting the file systems.

12. (Original) The apparatus of Claim 11 wherein the needed information is names of devices within which the file systems are located.

13. (Original) The apparatus of Claim 12 wherein the file systems are exported without first being mounted.

14. (Original) The apparatus of Claim 13 wherein the file is an extended attribute file.

15. (Original) The apparatus of Claim 14 wherein each mount point has an extended attribute file.

16. (Previously presented) A computer system for exporting file systems comprising:

at least one storage device for storing code data; and

at least one processor for processing the code data to consult a file associated with a mount point of a mounted file system to retrieve needed information to export the file systems, the mount point being the point at which the file systems are mounted on ~~a~~ the computer system, and to export the file systems.

17. (Original) The computer system of Claim 16 wherein the needed information is names of devices within which the file systems are located.

18. (Original) The computer system of Claim 17 wherein the file systems are exported without first being mounted.

19. (Original) The computer system of Claim 16 wherein the file is an extended attribute file.

20. (Original) The computer system of Claim 19 wherein each mount point has an extended attribute file.